AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method for dynamic striping, comprising:

    receiving a request to write a <u>first</u> data block into a storage pool;

    determining a physical disk location in the storage pool to store the <u>first</u> data block using a dynamic striping policy<u>, wherein the dynamic striping policy comprises at least one selected from the group consisting of a dynamic striping policy based on physical disk speed, a dynamic striping policy based on free space available on physical disks, a dynamic striping policy based on load on physical disks, and a round robin policy</u>;

    storing the <u>first</u> data block at the physical disk location; [[and]]

    storing a first indirect block in the storage pool, wherein the first indirect block comprises the data block location and the data block checksum<u>;</u>

    <u>changing the dynamic striping policy to obtain an updated dynamic striping policy; and</u>

    <u>storing a second data block using the updated dynamic striping policy.</u>

2. (Currently Amended) The method of claim 1, further comprising:

    retrieving the <u>first</u> data block using the first indirect block.

3. (Original) The method of claim 1, further comprising:

    assembling the first indirect block, wherein assembling the first indirect block comprises populating a block pointer.

4. (Currently Amended) The method of claim 3, wherein populating the block pointer comprises:

    storing the <u>first</u> data block checksum in a checksum field within the block pointer; and

    storing the <u>first</u> data block location in the block pointer, wherein storing the data block location comprises storing a metaslab ID and offset.

5. (Original) The method of claim 4, further comprising:

    storing a birth value in a birth field within the block pointer.

6. (Original) The method of claim 3, wherein the first indirect block is assembled using a data management unit.

7. (Original) The method of claim 1, wherein the storage pool comprises at least one storage device.

8. The method of claim 1, wherein the storage pool is divided into a plurality of metaslabs.

9. (Original) The method of claim 8, wherein each of the plurality of metaslabs is associated with a metaslab ID.

10. (Currently Amended) The method of claim 9, wherein the <u>first</u> data block location comprises the metaslab ID and an offset.

11. (Currently Amended) The method of claim 1, wherein storing the <u>first</u> data block comprises using a storage pool allocator.

12. (Cancelled)

13. (Currently Amended) A system for storing a <u>first</u> data block, comprising:

    ~~a storage pool comprising the data block and a first indirect block, wherein the first indirect block comprises a data block checksum and a~~ <u>~~first~~</u> ~~data block location;~~
    ~~and~~

    ~~a storage pool allocator configured to store the data block and the first indirect block in the storage pool using a dynamic striping policy;~~

    <u>a storage pool comprising:</u>

        <u>a plurality of child blocks, wherein each of the plurality of child blocks comprises one selected from the group consisting of the first data block and a first indirect block, wherein the indirect block references at least one of the plurality of child blocks;</u>

        <u>a parent block referencing at least one indirect block; and</u>

        <u>a storage pool allocator configured to store the parent block and the plurality of child blocks in the storage pool using a dynamic striping policy,</u>

        <u>wherein the dynamic striping policy is changed to obtain an updated dynamic striping policy, and</u>

> wherein the storage pool allocator is further configured to store a second data block in the storage pool using the updated dynamic striping policy.

14. (Original) The system of claim 13, further comprising:

a second indirect block, comprising a first indirect block checksum and a first indirect block location,

wherein the storage pool allocator is further configured to store the second indirect block in the storage pool.

15. (Original) The system of claim 13, further comprising:

a data management unit configured to assemble the first indirect block and request the storage pool allocator store the first indirect block.

16. (Original) The system of claim 13, wherein the dynamic striping policy comprises at least one selected from the group consisting of a dynamic striping policy based on physical disk speed, a dynamic striping policy based on free space available on physical disks, a dynamic striping policy based on load on physical disks, and a round robin policy.

17. (Original) The system of claim 13, wherein the storage pool comprises at least one storage device.

18. (Original) The system of claim 13, wherein the storage pool is divided into a plurality of metaslabs.

19. (Original) The system of claim 18, wherein each of the plurality of metaslabs is associated with a metaslab ID.

20. (Currently Amended) The system of claim 19, wherein the first data block location comprises the metaslab ID and an offset.

21. (Currently Amended) A computer system for dynamic striping, comprising:

a processor;

a memory;

a storage device; and

software instructions stored in the memory for enabling the computer system under control of the processor, to:

receive a request to write a <u>first</u> data block into a storage pool;

determine a physical disk location in the storage pool to store the <u>first</u> data block using a dynamic striping policy, <u>wherein the dynamic striping policy comprises at least one selected from the group consisting of a dynamic striping policy based on physical disk speed, a dynamic striping policy based on free space available on physical disks, a dynamic striping policy based on load on physical disks, and a round robin policy</u>;

store the <u>first</u> data block at the physical disk location; [[and]]

store a first indirect block in the storage pool, wherein the first indirect block comprises the <u>first</u> data block location and the <u>first</u> data block checksum;

<u>change the dynamic striping policy to obtain an updated striping policy; and</u>

<u>store a second data block in the storage pool using the updated striping policy.</u>

22. (Currently Amended) A network system having a plurality of nodes, comprising:

~~a storage pool comprising the data block and a first indirect block, wherein the first indirect block comprises a data block checksum and a~~ <u>~~first~~</u> ~~data block location;~~

~~and~~

~~a storage pool allocator configured to store the data block and the first indirect block in the storage pool using a dynamic striping policy,~~

<u>a storage pool comprising:</u>

<u>a plurality of child blocks, wherein each of the plurality of child blocks comprises one selected from the group consisting of a first data block and a first indirect block, wherein the indirect block references at least one of the plurality of child blocks;</u>

<u>a parent block referencing at least one indirect block; and</u>

5

a storage pool allocator configured to store the parent block and the plurality of

child blocks in the storage pool using a dynamic striping policy,

wherein the storage pool is located on any one of the plurality of nodes, [[and]]

wherein the storage pool allocator is located on any one of the plurality of nodes,

wherein the dynamic striping policy is changed to obtain an updated dynamic

striping policy, and

wherein the storage pool allocator is further configured to store a second data

block in the storage pool using the updated dynamic striping policy.